**Financial Consumer Complaint Classification**

Final Report

IST 718 Big Data Analytics | Spring 2024

Group 4

Francisco Franco Arenas

Qidu Quentin Fu

Yixing (Leo) Zhu

Ximeng Deng

**Project Overview**

The primary goal of this project is to develop a machine learning model that automatically classifies consumer complaints about financial products based on their text narrative and related attributes. It aims to enhance how financial institutions handle and respond to complaints, ensuring they are quickly directed to the right support department.

Financial institutions face challenges in manually processing the large volume of consumer complaints they receive daily. This manual sorting is time-consuming, costly, and error-prone, leading to delays and, consequently diminished customer satisfaction. An automated system is crucial for efficiently categorizing complaints and improving the resolution process.

Our solution involves the creation of a machine learning algorithm that uses natural language processing (NLP) to analyze and categorize complaint texts. This will allow financial institutions to automatically route complaints to the relevant department, optimizing the resolution workflow.

**Dataset Description**

The data comes from the Consumer Complaint Database of the Consumer Financial Protection Bureau. This database is a collection of consumer complaints about financial products and services that have been forwarded to companies for a response. Complaints are made public after a response from the company, confirmation of a commercial relationship with the consumer, or after 15 days, whichever occurs first.

The dataset is directly linked to a live database (https://catalog.data.gov/dataset/consumer-complaint-database), so it is constantly updated. For the purposes of this project, we have set a cutoff date of January 1, 2024; only data published before this date, which includes approximately 4.5 million rows, has been used.

The data consists mostly of text (13 out of 18 columns), which requires extensive text cleaning before it can be used in machine learning models. The proposed predictors for model building are Date Sent to Company, Sub-Issue, Company, Narrative, and Zip Code, with Product (complaint type) as the target variable.

We dropped duplicates, deleted rows where the Narrative contains more than 800 words, and selected 6 columns (variables) from the original dataset and rows where all 6 columns are not empty. Now, there are 1,442,917 rows.

**Data Exploration and Feature Engineering**

Text dataset cleaning was a crucial preprocessing step in our natural language processing project. Since the primary column was the complaint narrative column, we spent much time on cleaning it. Cleaning involved standardizing text to enhance the quality and effectiveness of analytics. Key techniques included tokenization, where text was split into words or phrases; normalization, which involved converting text to a uniform case; and stop-word removal, which eliminated commonly used words that offered little value in analysis. Additionally, lemmatization was used to reduce words to their base or root form, further homogenizing text and improving algorithmic performance. Effective cleaning not only streamlined subsequent processing but also significantly boosted the accuracy of our models by reducing noise and focusing on relevant information. Subsequent processing stages included tokenization, the use of CountVectorizer, and the application of Term Frequency Inverse Document Frequency (TF IDF), to adjust the importance of words based on how frequently they appear across all documents, thus highlighting more distinctive terms. Note that we tried to use Bidirectional Encoder From Transformers (BERT) bedding from Hugging Face and PySpark, which were not compatible with the machine learning pipeline built into PySpark. Therefore, TF IDF was finally used to represent our text data.

To further prepare the data for machine learning models, we further processed the data, specifically the target column. For the product column, we processed the categories that were part of a nested relationship and combined categories with very few rows into broader categories. Through human level thematic analysis of the issues column, we further grouped the issues and obtained a total of 5 categories for the classification model.

| Before |
| --- |
| After |
| Most Frequent Words across Product |
|  |
| Frequency Distribution of Word Counts, Frequency of Complaints per Year |
|  |

We examined the chronological distribution of complaint narratives in the dataset, which spans from 2015 to 2023. The number of complaints has generally increased over time, with a particularly noticeable surge from 2022 to 2023.We encoded month and day with cyclical transformations, applying sine and cosine functions to capture the cyclical nature of time in the features. This is because traditional coding would make January and December appear very distant, as well as the first and last days of the month. However, cyclical transformations represent these time periods in a way that places December and January next to each other, maintaining their natural continuity.

**Two-Stage Machine Learning Model**

**Indexing**

We built a two-stage machine learning model to handle the imbalanced categories within the “Product” column. Initially, we indexed all unique values in this column, assigning values 0 through 4 in the “Product\_index” column. Label 0 was designated for the majority class, which comprised 65% of the data. In the first stage of the model, we created a new column named “first stage index.” Here, the majority class retained the index 0, while minority groups—categories 1, 2, 3, and 4—were collectively re-indexed as 1 to differentiate them from the majority. For the second stage, to balance the classes, we randomly sampled 50% of the rows where “Product\_index” was 1.0, and included all instances from categories 2, 3, and 4.

**Results**

The best model we got was using the random forest algorithm. First, the model predicts whether an instance belongs to the most common class, Credit Reporting, Credit Repair Services, or Other Personal Consumer Reports. The training phase achieved a receiver operator characteristics (ROC) area under the curve (AUC) of 0.9108, and the testing phase reached a ROC AUC of 0.9114, with an overall accuracy of 78.19%. However, in terms of class-specific performance, the recall and specificity for the majority class (label 0) were both 0.9869, and for the minority class (label 1), the recall was 0.4029 and specificity was 0.9434.

During the second stage multi-class classification, the train F1 Score was 0.7635 and the test F1 Score was 0.7646. Finally, we merged both predictions and obtained a confusion matrix and got a F1 score of 0.6276. It was slightly more accurate in predicting Debit card or prepaid card, as well as Loan.

**Problems Encountered and Limitation**

We tried to incorporate BERT embeddings into our Spark NLP workflow for text data processing; however, the results were not satisfactory. Upon utilizing `small\_bert\_L2\_128` on a reduced sized dataset, we found that embedding the entire dataset was excessively time-consuming. Therefore, we chose to employ the TF-IDF method for this project. It is noteworthy that in our tests with reduced datasets, BERT embeddings did not get much better performance compared to the simpler TF-IDF approach.

**Summary**

Our results were better than the baseline by no-information predictions, which is acceptable, though not perfect. For future work, to address imbalances in the data, we can adjust weights to enhance the recall rate of samples from minority groups.